

----- Listing_4 -----

```
$crystal = 8000000           `deklaracja częstotliwości kwarcu
$regfile "m8def.dat"       `biblioteka mikrokontrolera
Deflcdchar 0 , 27 , 17 , 17 , 17 , 17 , 17 , 17 , 27
Deflcdchar 1 , 31 , 31 , 31 , 31 , 31 , 31 , 31 , 31
`definicja dwóch nowych znaków wyświetlanych podczas powitania
Config Pind.0 = Input
Config Pind.1 = Input
Config Pinc.5 = Input
Config Pinc.4 = Input
Config Pinc.3 = Input
Config Pinc.2 = Input
`deklaracje pinów jako wejścia (przyciski sterujące)
Config Pinb.4 = Output
Config Pinb.3 = Output
Config Pinb.2 = Output
Config Pinb.1 = Output
`deklaracje pinów jako wyjścia (diody sygnalizacyjne)
Set Portd.0
Set Portd.1
Set Portc.5
Set Portc.4
Set Portc.3
Set Portc.2
Set Portb.4
Set Portb.3
Set Portb.2
Set Portb.1
`włączenie wewnętrznych rezystorów podciągających
Config Lcd = 20 * 4         `deklaracja typu wyświetlacza
Config Lcdpin = Pin , Db4 = Portd.5 , Db5 = Portd.6 , Db6 =
Portd.7 , Db7 = Portb.0 , E = Portd.3 , Rs = Portd.2
`deklaracja pinów do połączenia z wyświetlaczem LCD
Display On           `inicjacja wyświetlacza
Cursor Off          `wyłączenie wyświetlania kursora
Cls                 `czyszczenie zawartości wyświetlacza
```

```

Lcd "Digital Power Meter"      `wyświetlamy napis
Locate 2 , 1                  `podanie lokalizacji na LCD
Lcd "portalnaukowy.edu.pl"    `i wyświetlenie tam napisu
Wait 3                        `czekamy 3 sekundy
Cls                           `czyścimy wyświetlacz
Locate 1 , 1
Lcd "Self Test Procesing"
Locate 2 , 7
Lcd Chr(1) ; Chr(0) ; Chr(0) ; Chr(0) ; Chr(0) ; Chr(0);
Portb.1 = 0
Wait 1
Locate 2 , 7
Lcd Chr(1) ; Chr(1) ; Chr(0) ; Chr(0) ; Chr(0) ; Chr(0);
Portb.2 = 0
Wait 1
Locate 2 , 7
Lcd Chr(1) ; Chr(1) ; Chr(1) ; Chr(0) ; Chr(0) ; Chr(0);
Portb.3 = 0
Wait 1
Locate 2 , 7
Lcd Chr(1) ; Chr(1) ; Chr(1) ; Chr(1) ; Chr(0) ; Chr(0);
Portb.4 = 0
Wait 1
Locate 2 , 7
Lcd Chr(1) ; Chr(1) ; Chr(1) ; Chr(1) ; Chr(1) ; Chr(0);
Portb.1 = 1
Portb.2 = 1
Portb.3 = 1
Portb.4 = 1
Wait 1
Locate 2 , 7
Lcd Chr(1) ; Chr(1) ; Chr(1) ; Chr(1) ; Chr(1) ; Chr(1);
Wait 1
Locate 3 , 1
Lcd "No errors found ..."
Wait 2

```

Cls

`powyższy fragment programu wyświetla pasek postępu testowania systemu i odpowiednie teksty, zapala też i gasi diody kontrolne. Funkcja raczej niepotrzebna ale ładnie się prezentuje ☺

Config ADC = **Single** , Prescaler = Auto , Reference = Avcc

`konfiguracja przetwornika

Config Timer1 = **Timer** , Prescale = 256 `konfiguracja timera

On Timer1 Cykl `realizacja podprogramu cykl po przepełnieniu

Enable Interrupts `globalne włączenie przerw

Enable Timer1 `włączenie timera

Timer1 = 59286 `wpisanie do timera początkowej wartości by po odliczeniu pozostałości uzyskać przerwanie co 0,2s

Dim Energy_flag **As Bit**

Dim Timer_flag **As Bit**

Dim 02sek **As Byte**

Dim Sek **As Byte**

Dim Minuty **As Byte**

Dim Godziny **As Byte**

Dim Volts1 **As Word**

Dim Volts2 **As Single**

Dim Volts3 **As String** * 3

Dim Sredniav **As Single**

Dim KrokV **As Byte**

Dim Current1 **As Word**

Dim Current2 **As Single**

Dim Current3 **As String** * 3

Dim Sredniac **As Single**

Dim KrokC **As Byte**

Dim Moc **As Single**

Dim Moc2 **As String** * 3

Dim Mocmax **As Single**

Dim Mocmax2 **As String** * 3

Dim Energy_chwil **As Single**

Dim Energy **As Single**

Dim Energy_wh **As Single**

```

Dim Energy_wh2 As String * 3
`deklaracja wszystkich potrzebnych w programie zmiennych
Start ADC          `uruchomienie przetwornika ADC
Mocmax = 0
Timer_flag = 0
Energy_flag = 0
`wpisanie początkowych wartości zmiennych
Do
Loop
End

`pusta pętla programowa
Cykl:      `poprogram realizowany po przepełnieniu timera co 0,2s
Timer1 = 59286
If Pind.0 = 0 Then
Timer_flag = 1
End If
`badanie stanu przycisku „timer start”, gdy naciśniemy flaga
timera jest ustawiana na 1
If Pind.1 = 0 Then
Timer_flag = 0
End If
`badanie stanu przycisku “timer stop”, flaga ustawia się na 0
If Pinc.5 = 0 Then
Gosub Zeruj_timer
End If
`badanie stanu przycisku “timer clear”, gdy naciśniemy
realizowany jest podprogram Zeruj_timer
If Pinc.4 = 0 Then
Energy_flag = 1
End If
If Pinc.3 = 0 Then
Energy_flag = 0
End If
If Pinc.2 = 0 Then
Energy = 0
Energy_wh = 0

```

```

End If
`analogicznie jak w przypadku timera
If Timer_flag = 1 Then
Portb.4 = 0
Portb.3 = 1
End If
If Timer_flag = 0 Then
Portb.4 = 1
Portb.3 = 0
End If
If Energy_flag = 1 Then
Portb.2 = 0
Portb.1 = 1
End If
If Energy_flag = 0 Then
Portb.2 = 1
Portb.1 = 0
End If
`obsługa diod LED sygnalizujących stan timera i miernika
Sredniav = 0
For Krokv = 1 To 10
Volts1 = Getadc(0)
Sredniav = Sredniav + Volts1
Next Krokv
Sredniav = Sredniav / 10
Volts2 = Sredniav * 0.048
`pomiar napięcia z uśrednieniem 10 następujących po sobie
pomiarów
Sredniac = 0
For Krokc = 1 To 10
Current1 = Getadc(1)
Sredniac = Sredniac + Current1
Next Krokc
Sredniac = Sredniac / 10
Current2 = Sredniac * 0.0089
`pomiar prądu analogicznie jak napięcia

```

```

Locate 1 , 1           `podanie lokalizacji na LCD
Volts3 = Fusing(volts2 , "##.&") `zamiana wyniku pomiaru
napięcia na tekst
Lcd "U= " ; Volts3 ; " V "      `wyświetlenie wyniku
Locate 1 , 11
Current3 = Fusing(current2 , "##.&")
Lcd "I= " ; Current3 ; " A "
`zamiana wyniku pomiaru prądu I wyświetlenie analogicznie jak w
przypadku napięcia
Locate 2 , 1           `Przejdźcie do drugiej linii wyświetlacza
Moc = Volts2 * Current2 `obliczenie mocy
Moc2 = Fusing(moc , "###.&")      `zamiana wyniku obliczenia na
tekst do wyświetlenia na LCD
Lcd "P=" ; Moc2 ; " W "          `wyświetlenie mocy
Locate 2 , 11          `przejdźcie na dalszą pozycję linii
If Moc > Mocmax Then
Mocmax = Moc
End If
`zatrzaśnięcie w zmiennej mocmax maksymalnej wartości mocy
Mocmax2 = Fusing(mocmax , "###.&")
Lcd "Pm=" ; Mocmax2 ; " W "
`zamiana na tekst I wyświetlenie szczytowej mocy
Locate 3 , 1 `Przejdźcie do linii trzeciej LCD
If Energy_flag = 1 Then
Gosub Energia
End If
`badanie stanu flagi miernika energii, jeśli „1” realizowany jest
podprogram „Energia”
Energy_wh2 = Fusing(energy_wh , "#####.&&&&")
Lcd "E= " ; Energy_wh2 ; " Wh"
`zamiana wartości energii obliczonej w podprogramie na tekst I
wyświetlenie jej w trzeciej linii.
Locate 4 , 1
If Timer_flag = 1 Then
Gosub Czasomierz
End If

```

'badanie stanu flagi czasomierza, jeśli "1" realizowany jest podprogram czasomierz.

Lcd "Time: " ; Godziny ; " h " ; Minuty ; " m " ; Sek ; " s "

Return

'wyświetlenie w czwartej linii obliczonych w podprogramie godzin, minut I sekund pracy.

Energia: 'podprogram obliczający energię chwilową i sumaryczną w czasie pracy.

Energy_chwil = Moc * 0.2 'E = P x t

Energy = Energy + Energy_chwil 'sumowanie energii

Energy_wh = Energy / 3600 'przeliczenie z Ws na Wh

Return 'powrót do programu głównego

Czasomierz: 'podprogram obliczający czas

Incr 02sek 'z każdym uruchomieniem tego podprogramu czyli co 0,2s zwiększamy wartość zmiennej 02sek

If 02sek = 5 **Then**

02sek = 0

Incr Sek

End If

'jeśli wartość zmiennej 02sek wyniesie 5 minie jedna sekunda i zwiększamy wtedy wartość zmiennej Sek, zerując jednocześnie 02sek.

If Sek = 60 **Then**

Sek = 0

Incr Minuty

End If

If Minuty = 60 **Then**

Minuty = 0

Incr Godziny

End If

Return

'analogicznie do zmiennej Sek postępujemy z pozostałymi.

Zeruj_timer: 'Podprogram uruchamiany po naciśnięciu przycisku timer clear. Zeruje wartości zmiennych czasu.

02sek = 0

Sek = 0

Minuty = 0

Godziny = 0

Return
